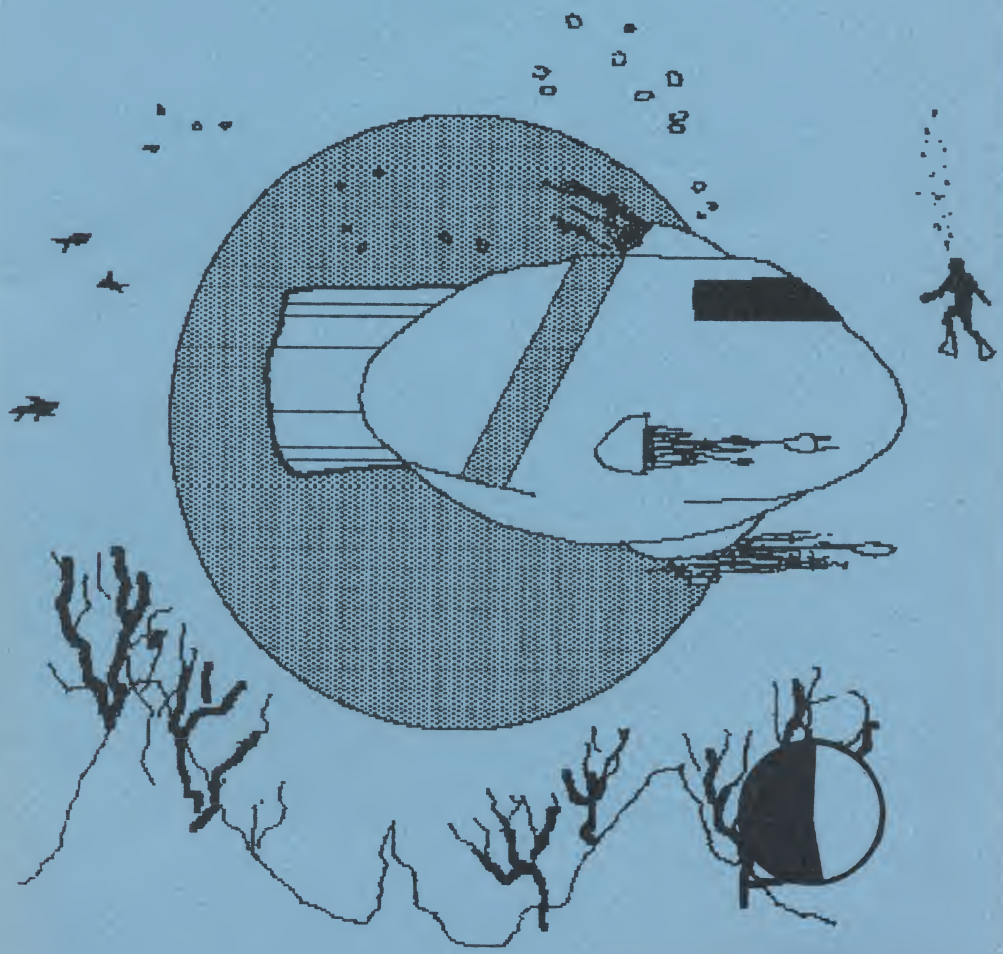
status

NEWSLETTER

VOLUME 5 - ISSUE 8 AUGUST - 1986



STATUS Officers

PRESIDENT

GENE RODRIGUEZ.....499-6021

VICE PRESIDENT

STAN HARRISON.....497-4369

TREASURER

DAVID LEVY.....623-7802

SECRETARY

DICK LITCHFIELD.....468-6964

PROGRAM COORDINATOR

BUCK MADDREY.....464-2100

LIBRARIAN

CHUCK SARGENT.....422-5691

MEMBERSHIP CHAIRMAN

BILL WRIGHT.....481-0555

SYSOP

DOUG BOYNTON.....468-6242

EDITOR

GENE RODRIGUEZ III.....490-2951

STATUS BBS

300/1200
804-468-1096

Editor's File

Some much to talk about and such a short space to say it in! What to get to first? Okay, the guilt meter.

You may or may not have noticed the little circle on the front cover, (If not, go check it out now) from now on that little symbol will be there to let you know what percentage of this Newsletter was derived from outside sources... do you feel guilty yet?

This issue marks the second anniversary of my tenure as editor of this fine publication (please hold your applause 'til the end) and I would like to say that even though there have been times that I've wanted to chuck the whole thing, I'm glad I've held out. I think it's been worth it...

Now for the **really** exciting stuff!

For years now, I've been going through the same process every month: collect/wrangle/mug a few articles from members, and then **off** I'd trudge to the Newsletter Exchange to spend a few painful nights typing in some articles to fill out the rest of *OUR* Newsletter. I always said to myself: "There has **got** to be a better way!"

Now there is.

Introducing **NewsNET!** The ATARI Electronic Information Network. Starting **August 4, 1986** NewsNET will be open to the public at **804-490-1348** from **7:00 pm to 7:00am (EST) Mon-Fri, and 24 hours on Sat & Sun.** This is how it works (pay attention all you editors, out there): what **NewsNet** will be, is a clearing house for Newsletter articles from across the country (and around the globe!), but it can only work if **everybody** helps out. So when you've put *your* publication to bed for the month, pop those files onto **NewsNET** and maybe pick some up for yourself for next month.

Set your modem to ASCII and give us a call, **NewsNET** has 20 MEG of storage for your submissions... right off the bat I can think of two advantages: 1) no more scurrying around for articles and typing your fingers to the bone and 2) almost guaranteed national exposure! Who could ask for more...

TELECOMPUTING

By Doug Boynton

Beep. Boop. Hello, again, telecommunicators! It's time for another discourse designed to massage your gray matter.

I had intended to flog the old "etiquette" horse again this month, but couldn't bring myself to write anything worth the ink it would take to print it with. So, moving right along, we'll make this sort of a potpourri of little bytes this month.

▩ The latest from CompuServe on the old, "Request recorded, one moment please" bugaboo is that yes, indeed, you **ARE** charged for the time spent staring at a blank screen while the big mainframes switch you among themselves. Customer Service had the following advice after a five-minute wait to get to SIG*ATARI the other night: "Call during non-peak hours". My response was as follows, "If *YOUR* system is overloaded, is that *MY* problem? Shouldn't you maybe cut back on those network TV spots to get new subscribers?" Still waiting for a reply on that one from Customer Service. I'm not holding my breath.

▩ Drop a line to your Senator. Specifically, Virginia's Junior Senator, **Paul Trible**. The name of the bill we're talking about here is the **"Computer Pornography and Child Exploitation Act of 1985"** (S. 1305). The senator wants to make sure child molesters don't use their computers to swap information, a laudable goal. The senator also wants to make sure Bulletin Board Systems are free from pornography. You see where we're headed here, right? What is pornography? If the courts can't decide, can a 15-year-old Sysop? Too much legislation pertaining to computer communication is born of ignorance. Modem users are getting a bad rap, and lawmakers are ready to impose massive restrictions on all of us... to halt the abuses of a few. Somehow... we've got to let the lawmakers know there are thousands of voters out here who use computer communications for legitimate business, professional, and hobby applications. We're not making enough noise. Speak up.

▩ It's fun to watch **GEnie** grow. General Electric's CompuServe alternative continues to spread its

wings; adding new sections all the time. The latest move pertaining to Atari users is the break-up of the **ATARI Roundtable** into **8-BIT** and ST groups. A wise move; beneficial to both groups. If you get an ST, be sure to start your library here. At $4.95/hour at 1200 baud, you'll save a lot of money over the "Request recorded..." company.

▩ 2400 Baud. You're going to be hearing more and more about it. Now that 1200 baud modems are under $100 in many cases, many folks are going to be taking a look at 2400 baud. The prices are still way, way up there, but many businesses are already moving to 4800 or 9600 baud (yes, over the phone lines!), and as that move continues, 2400 baud modems will hit the "used" pile, and become available to hobbyists at reasonable prices. CompuServe and GEnie have begun placing 2400 baud nodes around the country already.

▩ I'm not a big fan of **ANTIC magazine**. But you may want to take a look at the August issue. There are a couple of articles dedicated to telecomputing this month that are worth the price of admission; especially if you're thinking of buying a new modem, or have bought a new modem.

Next month, Dick Litchfield and I will try to put together a comparison of the modems that claim to be "Hayes-Compatible". After that... geez, I dunno. I'm sort of running low on ideas. How about some input from you? What would you like to see here? Instructional? Gossip and rumors? Legislative action? Philosophy? Unique home-spun humor? Drop me a line.

Is there a Language for you on the Atari ST?

By Wilfrod Niepraschk

Many people have asked me what language I use for writing programs. This is not an easy question to answer, but maybe if I describe a few of the languages available for the ST computer your choice will be a little easier to make.

BASIC

Basic has been around for a long time and is currently being used by about 85% of the people who have home computers. This lan-

guage is popular because it is very easy to use and it is one of the most powerful languages available for micro computers. Most Basics allow you to create screen formats and graphics with ease, while most other languages require you to write your own routines.

Basic is an interpreter, therefore it executes much slower than most of the other languages available. Program development is much easier with an interpreter since you can run a program and see instant results. If the program did not meet up to your expectations you may modify it and run it again without having to compile or link it each time. This link and compile time may take up to ten minutes for other languages. Basic compilers are available for some Basics that will take the interpretive code and compile it to object code, therefore giving you the speed of execution while still making it easy to develop programs.

The ST computer comes with **ST BASIC**. This basic has graphic support and most other functions normally found in that language. ST BASIC is similar to **Microsoft Basic**, but programs on an IBM PC will not run on the ST without major changes. File handling and cursor positioning are the biggest problems when trying to convert from the IBM PC computer. It sure would have been nice if ST BASIC would have been written to be 100% compatible, as that would have allowed the user to have access to thousands of programs.

ST BASIC is very slow and clumsy to use. The editor is the worst I have ever seen and the windows constantly get in my way. The GOTOXY function does not appear to be working properly all of the time. I have always been a great fan of BASIC but at the present I do not recommend using ST BASIC because of it's hostile environment.

Other Basics are available, but I would look at them very carefully before you spend your hard earned cash.

PASCAL

Pascal is used by about 30% of the programmers. It is very popular in universities as a teaching language. Pascal forces the programmer to write programs that are well structured. Since Pascal is a compiler the

programs written in this language will execute much faster than programs written in ST BASIC. Pascal requires a lot of extensions to make it suitable for writing business applications.

I purchased **Personal Pascal** from **Optimized Systems Software Inc.** (OSS). OSS has always put out fine products and their Personal Pascal adds just one more to their list. It features many extensions. String extensions are similar to the ones used in **TURBO PASCAL** from Borland. This package is very user friendly and it is the easiest user interface I have ever seen on any computer system. Once your program is written using the full screen editor the user needs only press a function key to save, compile and link that program. If an error is encountered during the compile, you may abort, continue, or return to the editor. If the last option was chosen then your file is automatically loaded and the cursor placed where the error occurred.

I found the editor very pleasing to use and wish that OSS would make it available as a separate item. The GEM interface is very easy to use. OSS decided to create their own GEM interface and they have done an exellent job of making it as painless as possible. The documentation is pretty good but the lack of an index makes it hard to find where everything is. I recommend this product to anyone that wants to write programs using GEM. You can write a nice looking program in very little time.

C

C is used by about 20% of the programmers. C is becoming very popular and many of the big applications are written in C. It is a very powerful and well-structured language that produces very compact object code. It is usually harder to learn C than Pascal or Basic and the source code may be a little harder to understand. You can change the name of keywords through the **DEFINE** statement, making the source almost impossible to read by someone not familiar with your concept.

I purchased the **ATARI DEVELOPMENT KIT** for $300 because it was the only one available at the time. This package included the **Alcyon C** from Digital Research. I was very disappointed with this product. While there was

about 1500 pages of documentation, it was almost impossible to find what you were looking for. The documentation talked more about the **IBM PC** than the **ATARI ST**. The user interface is nonexistent and compile and link time are about 10 minutes when using a floppy disk drive. There were also numerous bugs, but most of those have now been corrected. The only good thing from this package is that I am now a registered Atari Developer which gives me access to their SIG on **Compuserve**. There are other C packages available for the ST and the **LATTICE C** and **MEGAMAX C** have both had favorable reviews.

OTHERS

I have been told by a friend that the **Modula 2** package from TDI is an excellent product. I have not had a chance to review this package, but I really like Modula 2. It is basically a better version of Pascal since it was developed by the same person. Writing programs in Assembly language can be fun, but don't expect to write too many applications since it takes much longer to write code in. It also is much harder to transport Assembly code to another system. I use Assembly language only if speed is of great importance like in a sort program.

Happy News

By Rolly Herman

WAND Newsletter: June, 1986

The long awaited **Happy Enhancement** 7.0 has finally arrived. I waited a year and a half, but it was worth it. I had ordered and paid for the upgrade chip and disk for my 810 drive to make it compatible with the 1050 Happy, and to make the drives able to backup the latest software. Here is my report:

Installing the chip in the 810 was quite simple. It required opening up the drive and removing the side board. The metal shield on the sideboard had to be opened, the old Happy chip removed, and the new chip inserted in its place. Put everything back together, and that is all there is to it.

The chip and the Rev. 7.0 made the drives completely compatible using the Happy program. The same Rev. 7.0 disk works for both the 1050 and the 810. The 1050 did not need any new hardware. The multidrive option is really fantastic. It is possible to make two backup copies faster

make two backup copies faster than it previously took to make one. I have not had a chance to try out all the modes and parameters, but so far I was able to backup **SynCalc, Synfile+,** and **Scanalyser** which defied copying previously. The manual is an upgrade of the previous manual. Most of the material is the same, with the new Rev. 7.0 instructions added. I have always found the Happy manuals a little difficult to read, but the program is menu driven and very easy to use.

The Happy Enhancement is not only the best backup system. it is, in reality, the only one. There was a runner up called the **Archiver** which was fairly useful. However, the Archiver is now obsolete, and out of production. Another current system, the **Duplicator**, is so bad, so full of bugs, and lacking in backup ability that it should not even be considered.

Richard Adams, (younger brother of Scott Adams, of adventure game fame) is the mastermind behind the Happy hardware and software, He has kept his system current with repeated upgrades so that if new protection schemes are devised, he figures out ways to make the backups. In his manual, he states, "there is no lock that a mortal can make that another mortal cannot create the key to,.....". He also emphasizes that the very powerful Happy system is to be used to make backups only for personal use of executable Atari programs and not for Piracy.

For those that do not already have the Happy Enhancement, and are contemplating buying it, the installation involves inserting a whole little daughter board, but it does not require any soldering, and is fairly easy to do. An added advantage with the 1050 drive is that the drive will now have true double density capability, as well as the single and enhanced densities.

The Happy Enhancement for the **810** or **1050** sells for $149.95. It is not available for the **Indus, Trak, Percom** or other third party drives. If you have an Atari drive, and need backups, then get HAPPY!!

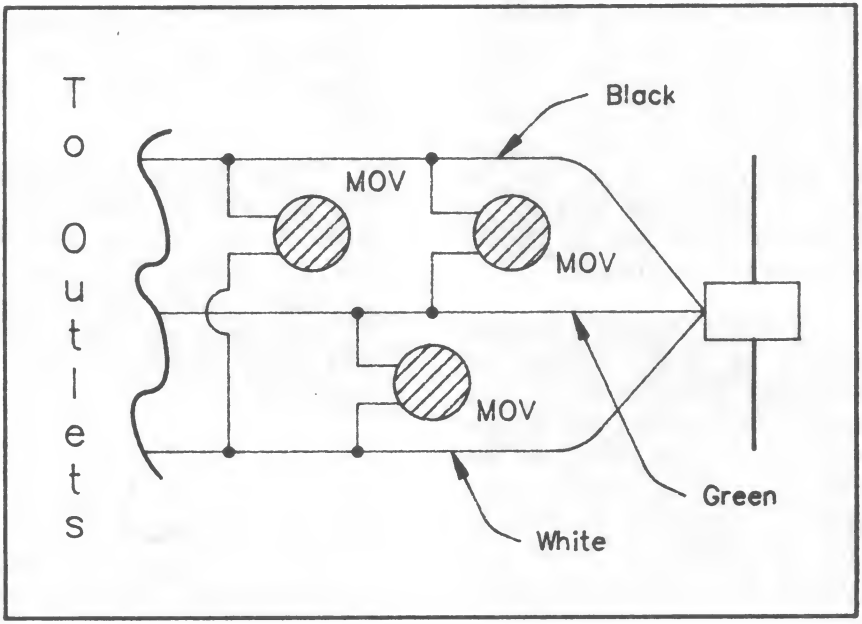
Upgrade Your Power Strip For Surge Protection

By Jim Parks

A multiple outlet power strip is one accessory that is common to almost every computer enthusiast, and by following the simple instructions here, you can upgrade your power strip to include surge protection that will equal those bought commercially.

The secret of surge protection is the **Metal Oxide Varistor** (or MOV for short). This device provides excellent protection against spikes and over voltages. Simply stated, once the voltage exceeds the MOV's rating, the excess voltage is shorted to ground.

To upgrade your power strip, start by unplugging and disassembling the case of your strip: most are secured with screws or rivets. Riveted cases may be drilled and later secured with sheet metal screws. At the point where the cord enters the case, the outer insulation of the cord should be skinned back exposing the black, white and green wires attached to the outlets. Scrape off a little insula-



tion from each of the three wires (carefully, don't cut the wires in two) and lay the power strip aside.

NOTE: The MOV's leads are not insulated, so plastic tubing or heat shrink tubing must be used on any exposed wires.

Study the diagram in Fig. 1 and solder the MOVs to the wires as shown. Then, with electrical tape, cover any bare wires and connections not protected by insulation. Carefully re-assemble the power strip so as to avoid pinching any wires. Your project is now complete, and you can enjoy the same peace of mind as you would with a commercially protected power strip.

PARTS LIST

3 - 130 Volt Metal Oxide Varistor's (**Radio Shack** 276-571 or equiv.)

Heat Shrink Tubing

Electrical Tape

ACTION!

By Mike Sawley

SLCC Journal: July, 1986

The **Action!** language by **Optimized Systems Software** is a non-portable language that generates very fast object code for the Atari **800** and **800XL** home computers. It is the fastest high level language available for these machines in both compilation and execution speed. It runs 100 times faster than Atari BASIC and executes more than twice as fast as Kyan Pascal.

Compilation time in Action! compared to Kyan Pascal is many times faster for several reasons. The Pascal compiler is large and requires relatively long load time, whereas Action! is already present on a ROM cartridge. Pascal makes type checks on 99% of the variable names and type declarations, while Action! guarantees type compatibility among its standard predefined types; no further type checking is done. Kyan Pascal uses a two pass compiler and Action! does it in one.

The design of the Action! system allows compilation from memory to memory, memory to disk and disk to memory while Kyan

Pascal compiles only from disk to disk due to the large size of the compiler. Action! even allows two different source codes (programs or subroutines) to reside in memory and can compile and run either from the monitor without building entire programs around them.

Many of the features of the prominent high level languages are found in Action! such as block structure, parameter passing, **IF-THEN-ELSE** nesting, record declaration and all of the high level loop constructs (no **GOTO** available!). Generally, if the feature contributed to the concept of structured programing while not slowing the execution of the object code (could be implemented with a minimum of output code)... the feature was put in. Other features must be defined if they are required.

It has been said that the construction of the Action! language was explicitly designed to optimize the **6502** instruction set and that is the reason for its fast execution speeds. That must be why the object code generated by Action! reads like that of a 6502 assembly language programmer (without the comments, of course!).

One thing that I found very discouraging was that recursion does not return popped values. Parameters are overwritten if called by the same subroutine. This is because a block of memory is used explicitly for variables and the names of the callers. The one pass compilation is another factor in this limitation. Recursion can be simulated, so I am told, and I am looking for a demonstration of this useful technique. Action! uses only the 256 byte hardware stack provided for the 6502 microprocessor to save return addresses from procedure and function calls during the run. While this is a factor in reducing the speed of execution, some of the utility of the language has been compromised.

The standard types supported are: **BYTE** (8-bit), **CARD** (16-bit used as addresses or large positive values), **INT** (-32768 - 32767), **CHAR** (256 characters in the set) and **POINTER** (for passing by reference etc.). Any coercions desired can be made between any of these standard types even though the lengths vary between 8 and 16 bits. **FLOAT** is an option which is

not included in the basic language but can be added at extra cost. This means that division truncates the decimal part. **ARRAY** is a standard structure and always numbers from 0 to n-1. It can contain any of the standard types. A special **TYPE** declaration can be made to declare records of standard types, But arrays of records must be implemented using a template which is moved over a larger array calculated in advance by multiplying the record size (in bytes) times the number of records. In this case, aliasing is a necessary evil. Dynamic array length declaration may not be done, but array bounds are not checked during execution. In fact, array length need not be specified at all. However, the user must insure that data does not collide with the program if this option is used.

Assignment in the type declaration is allowed. However, if the value is not enclosed in square brackets, the assignment is assumed to be binding of an address to the variable name itself. Also, arrays may be bound to any feasible starting address by the programer using the same technique. Since overflow is not checked, any values assigned during compilation or execution will be made modulo 256 or 65536 depending on the type of the variable being assigned to.

Overloading of operators is generally not done, but Action! has some other strange peculiarities. There are some operations which have more than one symbol to mean the same thing! "Not Equal" can be represented by either <> or #. "And" can be either AND or &. "Or" either OR or %. "Exclusive-Or" either XOR or !. And BYTE is a CHAR is a BYTE? All of this interchangeability is logical, but strange when the size of the language was originally limited because of speed considerations.

Procedures and functions that return values are an important part of the language. Their form is somewhat similar to Pascal except that nesting of them is not allowed. Any subroutine calling another must call above due to the one pass restriction. The reserved word RETURN is used at the bottom of procedures and functions to indicate their termination. A missing RETURN is not a compiler error. The BEGIN, END and

semicolon are not used as delimiters. Instead, an EOLN separates each line logically. The semicolon is used, as in Assembly Language, to indicate that a comment follows.

Eight parameters maximum may be passed to a subroutine. This is a violation of the "Zero-One-Infinity" Rule, but microcomputers do have limitations. Also, there is a restriction to not use channel 7 for input/output as that is reserved for the keyboard only by the Action! system.

Over all, the language is designed for the use of experienced programmers who need the flexibility to manipulate the operating system and the ability to write fast, readable code. Then, the full capability of the real-time graphics on the 8-bit machine can be realized.

Educational Computing Graphics

By Bob Moss

BAAUG Newsletter: July, 1986

When we bought our ATARI 800 more than four years ago, one of the major factors was the excellent graphics and color provided by the ATARI. This is still true. Even the old 8-bit ATARI computers have better graphics and color capability than the IBM PC and APPLE Macintosh. A major selling point of the Amiga and the ATARI ST series computers is the outstanding graphics.

Graphics programs available for micro computers range from color doodling to sophisticated computer aided design. There are many applications available costing under $100. My wife questions the educational value for children of most graphics programs, but I look at them as similar to basic art and design classes which allow a wide range of experimentation with composition and the relationship between elements. Which programs and input method is best depends on the abilities and interests of the child as well as the features of the program.

There are five ways to use graphic programs: mouse, joystick, graphics pad, light pen and keyboard. Since I am limiting this discussion to 8-bit ATARIs, I will exclude mouse input systems. One of the simplest joystick controlled programs is **Video**

Easel. Drawing is simple, almost like sketching or doodling. Younger children seem to like this, because they can make pictures easily. There are a variety of effects, colors and patterns possible, and good hand-eye coordination is not needed. **Paint** is another excellent graphics program for preschoolers and young children. A variety of colors and patterns is possible, and the techniques are easy for young children to learn. **Fun With Art** is a joystick controlled program similar in design to Paint. It offers a variety of features, such as setting colors to overwrite or be overwritten by other colors while drawing. Our experience is that it was too complex for children under 7 or 8 to really enjoy, though it has many nice features.

Micropainter is controlled by both joystick and keyboard. This program is more sophisticated, and offers more variety and interest. It also requires more skill and understanding of a variety of commands. Widths of lines can be changed, shapes can be drawn easily, there are fill and paint functions and colors and tints can be varied. It also is more difficult for younger children to understand all of the features, and to read and follow the directions. The added features of programs such as this offer children in the 7 to 10 age group an opportunity to experiment with art and graphics at a comfortable level. **Graphics Magician** is a rather sophisticated program which is joystick controlled with some keyboard commands. One interesting feature is the ability to mix text and graphics; another is that the memory stores the drawing sequence, not the final drawing, so that copying or reviewing the drawing recreates the actions of executing the drawing. This program would be used more effectively by older children, perhaps 10 to 16 year olds. All of these programs allow a wide range of colors and tints, drawing with different width lines, making repeatable geometric figures, storing and loading pictures from a screen dump and editing or changing the drawing on the screen. These programs are similar in concept and operation, and all have excellent features, but they probably will not be equally successful in appealing to younger children. All of them should be tried

by the child before buying, if at all possible.

We have two programs which use light pens: **Atarigraphics** and **Koala** lightpen. Both programs have pop-up menus, and have features similar to Paint or Fun With Art. Drawing directly on the screen is more natural than using a joystick or mouse, but it also limits the user mostly to preselected commands and graphic entry mode. We found them more of interest to older children, 8 to 12, than to preschoolers or 1st graders. I prefer the Koala lightpen, because the commands are more graphically described on the screen, and are easier to understand and apply than Atarigraphics. It takes some dexterity to use these programs, though young children can master the most of the commands and features after 1 or 2 lessons.

Graphics pads allow the user to draw on a flat surface and have the drawing reproduced on a screen to be saved, stored or shown as artwork. The 2 major graphics pads for the ATARI are **Koala** and **Chalkboard**. Koala has a smaller surface area. It has good graphics software, similar in many ways to the Koala lightpen program. The major disadvantage is that there are only a few programs for the 400/800 computers. Also, the small surface area may not be large enough for younger, less coordinated children. Koala has not been new ATARI programs for the Koala pad recently. Chalkboard is a much larger pad, making it more forgiving of drawing errors. There are a number of good programs available for Chalkboard. Our preschoolers and younger children actually prefer some of the non-graphic programs, such as **Bearjam** and **Music Maestro** rather than the **Microillustrator** graphics program, which is very similar to the Koala program. **Leo's 'Lectric Paintbrush** is a much better graphics program for younger children. The major problem with Chalkboard is that it has serious financial problems, has been in Chapter 11 bankruptcy since 1985, and may not be available.

A good graphics program is an excellent way to introduce younger children to computers. Some type of graphics program should be part of every ATARI educational software library.

New Product Info

[**Editor's Note**: All right, I've succumbed to the pressure I've felt from other Newsletters to print *something* about the summer CES, so here it is...]

Artworx is showing its bridge programs for both 8-bit and ST, and **Hole-in-One Golf** for the ST. They also have a new 8-bit program called **Peggammon**...

Omnitrend: **Universe II** for the ST...

First Star Software: the company that made **Spy vs. Spy** and **Boulder Dash** for the 8-bits, is coming out with **Comic Strip Maker**, a graphics program for the ST...

ICD: some exciting 8-bit hardware, including the **ICD Multi I/O Board**, a new multi-use I/O board that plugs into the parallel port...

MicroProse: ST version of **Silent Service**...

Zobian Controls: **The Rat**, a mouse for 8-bit Atari computers...

Batteries Included: **Thunder**, a real-time spelling checker, and a new version of **PaperClip** for the 8-bit with **Spellpack**...

Quickview: Software author Paul Heckel showed his **Zoomracks** ST database program...

Covox: **Voice Master**, Atari 8-bit voice recognition software...

Spinnaker: an extensive line of 8 and 16-bit educational and adventure game products...

Avila Associates: **Make it Move**, an ST graphics animation program, and a gambling tutorial currently called **Casino Craps**...

Britannica Learning: a series of educational programs for the 8-bit machines...

OSS: 8-bit and 16-bit programmer's tools...

[I guess that hits the high points, besides: it filled up the space left by our President who was unable to submit his monthly dose of wit, humor and insight due to the press of business... **Ed.**]

Opinions expressed in this publication are those of the individual authors and do not necessarily express or reflect the opinions of the Southside Tidewater Atari Technical Users Society (STATUS). Some material contained herein may have been taken from Bulletin Board Systems and/or Newsletters of other groups and should not be construed as fact.

Please address all Newsletter correspondence to:

STATUS
Newsletter Exchange
4836 Honeygrove Road
Virginia Beach, VA 23455
(804) 499-6021

Meetings: STATUS meetings are held on the first and third Thursdays of the month at the 7-UP Bottling Company, 5700 Ward Avenue, in Virginia Beach at 7:30 p.m. All interested parties are welcome to attend.

Newsletter Articles:

Submitted articles are preferred as disk text files, but will be gratefully accepted as hard copy (including handwriting) if you do not have a disk drive. If you have a modem, you can upload your articles to the Editor by calling the STATUS BBS at 468-1096. Articles may be submitted at any time, but will probably not make that month's Newsletter if submitted less than one week prior to the first meeting of the month.

S.T.A.T.U.S
Newsletter Exchange
C/O Gene Rodriguez
4836 HONEYGROVE RD.
Virgina Beach, VA 23455

First Class Mail